REMARKS

Claims 1-11 are pending in the application. Claims 1, 4, 5, and 7-11 are amended above. New claim 12 is added above. No new matter is added by the claim amendments or new claim. Entry is respectfully requested.

The abstract is objected to for reasons stated in the Office Action. The abstract is amended above in a manner that is believed to be consistent with suggestions made in the Office Action. Removal of the objection is respectfully requested.

Claims 4 and 8 stand objected to for informalities stated in the Office Action. The claims are amended above in a manner that is believed to be consistent with suggestions provided in the Office Action. Entry of the amendments and removal of the rejections are respectfully requested.

Claims 8 and 9 stand rejected under 35 U.S.C. § 112, second paragraph, for reasons stated in the Office Action. The claims are amended above in a manner that is believed to overcome the rejections. Reconsideration and removal of the rejections are respectfully requested.

Claims 1-11 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Talcott (U.S. Patent No. 6,272,623). It is respectfully submitted that this rejection be reconsidered and removed in view of the foregoing amendments and the following remarks.

The present invention of amended independent claim 1 is directed to a branch predictor for a multi-processing computer able to execute multiple processes, each process having a designated process reference. A history register stores a branch history of previous sequential branch instructions. A hash logic creates an index from a combination of a process reference of a process corresponding to a current branch instruction, an address of the current branch instruction, and the branch history. A branch prediction table stores branch prediction reference data and outputs branch prediction reference data corresponding to the index created by the hash logic. An address selection circuit selects one of a target address known from the current branch

instruction and a next instruction of the current branch instruction to generate a branch prediction address in response to the branch prediction reference data output from the branch prediction table. A branch prediction result tester updates the branch history stored in the history register and the branch prediction reference data stored in the branch prediction table in response to a real branch address and the branch prediction address according to the execution result of the current branch instruction.

The present invention of amended claim 10 is directed to a method of predicting a branch address of a conditional branch instruction with reference to a branch prediction table for storing branch prediction reference data in a multi-processing computer able to execute multiple processes, each having a designated process ID. An index is created to access the branch prediction table from a combination of a process ID of a process corresponding to the conditional branch instruction, an address of the conditional branch instruction, and a branch history comprising previous sequential branch instructions. Branch prediction reference data is read from the branch prediction table in response to the index. One of a target address known from the conditional branch instruction and a next address of the conditional branch instruction is selected in response to the branch prediction reference data. The branch history and the stored branch prediction reference data are updated in the branch prediction table in response to a real branch address according to an execution result of the conditional branch instruction.

The present invention of amended independent claim 1 therefore includes a "hash logic" that creates "an index" from a "combination of a process reference of a process corresponding to a current branch instruction, an address of the current branch instruction, and the branch history" (emphasis added). The "branch predictor" of the present invention of claim 1 is for a "multi-processing computer able to execute multiple processes, each process having a process reference" (emphasis added). The "process reference" used to create the "index" in the "hash logic" is therefore derived from the process corresponding to the current branch instruction. Such a "process reference" is illustrated, for example, as a 4-bit process ID in Table 1 at page 10 of the present specification. In this example, the process ID (ID) is combined with the branch

history (GH) and the address of the branch instruction (PC) to generate a hash index value (HI). The combination given in the example takes the form of an exclusive-OR operation (specification, page 10, lines 8-10).

An index for accessing the branch prediction table is therefore generated based on the combination of "the process reference", the "address of the current branch instruction", and the "branch history". The information contained in the branch prediction table is not flushed, and is therefore maintained, at the time of task switching. Therefore, the present invention provides for a high branch prediction rate, even in the case of frequent task, or context, switching where various processes are simultaneously active in the microprocessor. In this manner, the hit ratio of branch prediction is improved in a multi-process environment and pipeline stall is reduced, which in turn provides for a reduction in program execution time. (Specification, page 16, line 2 - page 17, line 1).

Similarly, the present invention of amended independent claim 10 comprises "creating an index to access the branch prediction table from a combination of a process ID of a process corresponding to the conditional branch instruction, the address of the conditional branch instruction, and a branch history comprising previous segmented branch instructions" (emphasis added). The "process ID" of claim 10 is again derived from the corresponding one of the "multiple processes" of the "multi-processing" computer, and the "process ID" contributes to the creation of the "index" that is used to access the branch prediction table.

It is submitted that Talcott fails to teach or suggest the present invention as claimed in claims 1 and 10. Specifically, with regard to claim 1, it is submitted that Talcott fails to teach or suggest "a hash logic for creating an index from a combination of a process reference of a process corresponding to a current branch instruction, an address of the current branch instruction, and the branch history" (emphasis added). While Talcott is cited in the Office Action at page 5, part b, as "creating an index" from a "combination" that includes "process references corresponding to a current branch instruction", detailed review of Talcott reveals that no mention

is made in Talcott as to branch prediction in a multi-process environment. Nor is there any teaching or suggestion in Talcott of distinguishing the process from which the branch instruction was derived through use of a "process reference". Nor is there any teaching or suggestion in Talcott of a "hash logic for creating an index" that is based in part on such a "process reference" as claimed in amended independent claim 1. Since Talcott is not directed to a multi-process environment, there is no need to distinguish the process that may be the source of a branch instruction. For this reason, there is no "process reference" in Talcott that contributes to the hashing function that leads to a branch prediction.

Similarly, it is submitted that Talcott fails to teach or suggest the present invention as claimed in amended independent claim 10. Specifically, there is no teaching or suggestion in Talcott regarding the creation of "an index" from a combination that is based in part on a "process ID of a process" in a multi-process environment.

In view of the above, it is submitted that Talcott fails to teach or suggest the present invention as claimed in amended claims 1 and 10. Removal of the rejections and allowance of independent claims 1 and 10 are therefore respectfully requested. With regard to the various dependent claims, it is respectfully submitted that these claims should inherit the allowability of the independent claims from which they depend.

Closing Remarks

It is submitted that all claims are in condition for allowance, and such allowance is respectfully requested. If prosecution of the application can be expedited by a telephone conference, the Examiner is invited to call the undersigned at the number given below.

Respectfully submitted,

Date: March 24, 2004

Mills & Onello, LLP
Eleven Beacon Street, Suite 605
Boston, MA 02108
Telephone: (617) 994-4900, Ext. 4902
Facsimile: (617) 742-7774
J:\SAM\0192\amendmenta2.wpd

Anthony P. Onello, Jr.
Registration Number 38,572
Attorney for Applicant